

# Robust Worst-Case Analysis of Demand-Side Management in Smart Grids

Javier Zazo, *Member, IEEE*, Santiago Zazo, *Member, IEEE*, and Sergio Valcarcel Macua, *Student Member, IEEE*

**Abstract**—Demand-side management presents significant benefits in reducing the energy load in smart grids by balancing consumption demands or including energy generation and/or storage devices in the user's side. These techniques coordinate the energy load so that users minimize their monetary expenditure. However, these methods require accurate predictions in the energy consumption profiles, which make them inflexible to real demand variations. In this paper we propose a realistic model that accounts for uncertainty in these variations and calculates a robust price for all users in the smart grid. We analyze the existence of solutions for this novel scenario, propose convergent distributed algorithms to find them, and perform simulations considering energy expenditure. We show that this model can effectively reduce the monetary expenses for all users in a real-time market, while at the same time it provides a reliable production cost estimate to the energy supplier.

**Index Terms**—Load management, robust analysis, non-convex optimization, distributed algorithms, game theory

## I. INTRODUCTION

SMART grids represent the concept behind the intended evolution of the electric grid, which through information acquisition from the end users will allow for more efficient energy distribution, flexibility in network topology, adaptive load management and better integration of renewable energy plants with the consumption requirements of the users.

One important concept to achieve the smart grid's objectives is the demand-side management, which includes the techniques for better energy efficiency programs, distributed energy generation, energy storage and load management. The model presented in this work establishes a robust energy price one day ahead, which takes into account the production costs of energy as well as uncertainties in the expected energy loads. We also consider a real-time market, where users are charged their specific demanded energy at the robust price, permitting some deviation without any extra penalty. This model provides some flexibility to every user in their real-time consumption energy loads and calculates the energy prices in a conservative manner, taking into account production costs and possible errors. In particular, we focus on a robust analysis of the demand-side management algorithms, where the energy strategies account for worst-case error deviations

in the predicted loads. In our formulation, these error terms vary from user to user and depend on the energy load profiles.

We show that this model can effectively reduce the monetary expenses of all users in the real-time market, while at the same time it can provide a confident production cost estimate to the energy supplier. Our analysis also shows that the added computational cost of these error calculations is minimal and the monetary gain is significant. Moreover, the calculations are amenable to a distributed implementation within the communications network of the smart grid. This is important because distributed algorithms are beneficial in terms of communication efficiency and computation scalability [1].

To further support our results we perform simulations in a real-time modeled scenario. We also compare our robust proposal with a naive version where users do not take into account price deviations, but where the energy supplier still establishes a robust price. Finally, we also present results illustrating the convergence behavior of our algorithm.

This study can be of special interest to energy supply authorities who can analyze how much energy they should account for if they plan for grid load estimation errors and peak demand costs. Even though users may be only interested in their monetary expenditure as energy consumers, supply authorities have to account for local deviations in the distribution network and coordinate with all agents to provide unobstructed service by controlling regular and peak supply stations. The fact that an agent in the network can alter the expected energy load by injecting or subtracting energy unexpectedly is a source of extra costs that have to be taken into account by suppliers that monitor the grid's energy demand. Therefore, these energy provision costs justify a modified pricing model and a worst-case analysis of the demand-side network.

### A. Related Work

The work presented here is focused on the study of the demand-side management framework through noncooperative game theory. We build our results from [2], where the authors consider a day-ahead scheduling algorithm and the users optimize energy generation and storage profiles in a distributed manner. Our model extends [2] by adding error terms to the expected consumption profiles (energy loads in the smart grid) and admitting variations in the estimated values with a bounded error. In addition, we analyze the effect that these error terms produce in the expected energy demand. In order to do so, we extrapolate some techniques from reference [3], where a robust algorithm for communications is developed.

Regarding a real-time energy market, several models have been proposed in the literature. For instance, the authors

All authors are affiliated with the Universidad Politécnica de Madrid, in Av. Complutense 30, Madrid, 28040, Spain.

Emails: javier.zazo.ruiz@upm.es, {santiago,sergio}@gaps.ssr.upm.es

This work was supported in part by the Spanish Ministry of Science and Innovation under the grant TEC2013-46011-C3-1-R (UnderWorld), the COMONSENS Network of Excellence TEC2015-69648-REDC and by an FPU doctoral grant to the first author. Paper no. TSG-00815-2015.

Digital Object Identifier 10.1109/TSG.2016.2559583

from reference [4] propose a real-time cost function with penalization terms in which the users minimize the expected monetary expenses. The energy production model assumes that the cost of energy is given by a quadratic function of the total demand, and therefore, the whole energy price of the smart grid is affected by the real-time variations. In such setting, the energy price is predetermined one day ahead, but it does not account for the unpredictable production costs of real-time demand. The penalty functions should account for these extra expenses, but they are also predetermined in advance. If they are set too costly, they would affect greatly the monetary expenses of users in the real-time market. On the other hand, if the penalties are set too low, they would cause an economic loss to the energy supplier. Our robust model, on the contrary, sets up an energy price that will allow the energy supplier to produce energy without incurring unplanned extra costs, and will allow users flexibility in their energy demand.

Another real-time model is introduced by reference [5], which proposes an optimization problem in which users demand energy according to a personal utility function. In this scenario prices are updated according to production costs, but demand is fully elastic, within some predetermined bounds, and the problem optimizes a social equilibrium point, rather than accounting for individual energy requirements. Furthermore, the energy supplier may experience monetary losses since the pricing mechanism does not cover production costs. In contrast, our algorithm accommodates individual energy requirements and ensures that production costs are covered by the pricing mechanism.

The authors of reference [6] propose a robust scenario for a real-time market based on linear programming, which allocates energy load profiles within a time frame. Because of the uncertainty of prices in real-time, users decide when to consume energy according to some modeling parameters, price range estimates and energy requirements. Although the algorithm is robust in the sense that it assumes a range of real-time energy prices, the model considers that prices are given by the energy supplier without taking into account production costs. Our robust model, on the other hand, establishes the prices one day ahead considering possible worst-case real-time variations, and takes into account production costs.

A relevant survey describing some of the last results and problems of the smart grid infrastructure can be found in [7], which analyzes different contributions and open problems in the topic, where demand-side-management is one example of these. Another publication that models the interaction among resident users and distributed energy resources is the one introduced by [8], but it considers prefixed energy generation and storage load profiles, as opposed to our framework, which can adapt the profiles depending on the energy prices. Regarding user behavior, the authors of reference [9] propose a model that takes into account the subjective behavior of users, but still does not consider imprecise load predictions.

Therefore, our contribution lies in considering a day-ahead energy market and minimizing the energy costs of users in a robust situation where energy demand errors are taken into account. In Section II we describe the demand-side model, the day-ahead cost function, the real-time market and briefly

describe the day-ahead optimization process. In Section III we introduce the robust game and analyze its solution through an equivalent formulation. We also present a distributed algorithm that solves the game and analyze its convergence properties. Finally, in Section IV we present simulations to illustrate the validity of our results.

## II. SMART GRID MODEL

In this section we introduce the smart grid model, the types of users in the demand-side of the network, the energy model, the pricing plan and the error terms. We consider a general framework with arbitrary convex regions and convex objectives.

### A. Demand-Side Model

The smart grid topology can be divided in three parts, namely the supply-side, a central unit or regulation authority, and the demand-side. The supply-side involves the energy producers, as well as the distribution network. The central unit coordinates the optimization process. The demand-side entails the individual end-users as consumers, possibly with means to generate and/or store energy and/or vary their consumption profiles. The demand-side may also include users with great impact on the smart grid, such as industries with high energy demands or opportunistic agents who participate as local energy suppliers in the network.

In this paper, the demand-side management focuses on calculating optimal policies for monetary savings on the consumers' side, while accounting for worst-case deviations from the consumption profiles due to unpredictable considerations in the demand-side. The optimization process involves all users together with the central unit, who exchange information for a more efficient use and distribution of energy. This communication is accomplished with smart meters, which are the devices in charge of the bidirectional transmission of data with the central unit and the optimization process.

We assume that all end users indicate their intended amount of energy consumption one day ahead, so that energy prices can be calculated based on demand levels during the optimization process. Then, the users can readjust their generation, storing and consumption profiles according to the variability of prices. This process is repeated iteratively until convergence. Algorithms that analyze this process have been studied in [2] and [10]. However, due to errors in their estimations, the real consumption may vary from the announced values. If these variations are uncorrelated among users, sometimes the energy left unused by some consumers will be spent by others inducing that, on average, the supply-side will not require to adjust their generation rates. However, this expected result may not always be unconditionally true. In addition, there is interest from the supply-side to provide energy for unexpected events that may require higher demands. Therefore, we present analytical results on identifying worst-case energy errors, for a given confidence interval.

We represent the set of users who participate in this process with  $\mathcal{D}$ , and additionally subdivide them into the set of active users  $\mathcal{N}$  and the set of passive users  $\mathcal{P}$ . Active users have some means of generating energy on demand (from dispatchable

sources), possess devices to store energy for future use or adapt their consumption profiles. On the contrary, passive users do not adjust their energy consumption profiles and their consumption is estimated from historical data.

The consumption values are determined one day ahead among all users, and are subdivided in time-slots  $h \in H = \{1, \dots, |H|\}$ . We define the *estimated per-slot energy load profile*  $l_n(h)$  that indicates the estimated energy usage of user  $n \in \mathcal{D}$  at time-slot  $h$ . We also represent these variables as row vectors  $\mathbf{l}_n = [l_n(1), \dots, l_n(|H|)] \in \mathbb{R}^{|H|}$ ,  $\forall n \in \mathcal{D}$ . Additionally, let  $\mathbf{l} = [\mathbf{l}_1, \dots, \mathbf{l}_{|\mathcal{D}|}] \in \mathbb{R}^{|H| \times |\mathcal{D}|}$  denote the global vector with the energy loads of all users. Each individual term  $\mathbf{l}_n$  includes the energy consumption of user  $n$  and the energy contributions or added expenditures that may decrease or increase the energy load due to the user owning some device to generate or store energy, respectively. Naturally, these profiles are variables to the optimization process and will vary according to the different strategies the user can select. It is satisfied that  $l_n(h) > 0$  if the energy flows from the grid to the user, and  $l_n(h) < 0$  otherwise, when the user is selling energy to the grid.

We assume that the total load in the network—*aggregate per-slot energy load*—at time slot  $h$  denoted by  $L(h)$  is positive:

$$0 < L(h) \triangleq \sum_{n \in \mathcal{D}} l_n(h) \quad (1)$$

and it is calculated by the central unit by aggregating everybody's estimated load, including passive and active user's load.

Finally, we introduce the *per-slot energy load error profile*  $\delta_n(h)$  as the difference from the estimated *per-slot energy load profile*  $l_n(h)$  and the real value. This difference comes from the assumption that demand-side users do not know their consumption requirements with precision and they may deviate from their estimates. Additionally, active users may also deviate from their calculated generation and storing policies for unpredictable reasons such as malfunctioning, disconnections or lack of collaboration. For these reasons, introducing an error term makes the model more realistic. We represent these error terms both as row vectors  $\delta_n = [\delta_n(1), \dots, \delta_n(H)]$ ,  $\forall n \in \mathcal{D}$  as the error profiles of user  $n$  and as column vectors  $\delta(h) = [\delta_1(h), \dots, \delta_{|\mathcal{D}|}(h)]^T$ ,  $\forall h \in H$ . Note that the index  $n$  or  $h$  allows to easily distinguish which one we are referring to. Additionally, let  $\delta = [\delta_1, \dots, \delta_{|\mathcal{D}|}] \in \mathbb{R}^{|H| \times |\mathcal{D}|}$  denote the global vector with the error terms of all users.

### B. Day-ahead Cost Model

The purpose of our day-ahead model is to establish an estimate of the energy price that is robust against real-time energy demand variations. From a practical perspective, it determines a worst-case energy production cost taking into account expected user energy loads plus uncertainties, so that the energy supplier does not suffer monetary losses due to unplanned real-time demand.

The energy price in the network is determined by the *grid cost function*  $C_h$ , which is fixed by the supply-authority and depends on the aggregate per-slot energy load  $L(h)$  at each time-slot. We use a quadratic cost function, which is widely used in the literature (see e.g. [2], [10]):

$$C_h(L(h), \delta(h)) \triangleq K_h \left( L(h) + \sum_{n \in \mathcal{D}} \delta_n(h) \right)^2 \quad (2)$$

where the coefficients  $\{K_h\}_{h=1}^{|H|} > 0$  represent the energy cost at each time slot determined by the supply-side. In addition, the individual cost for user  $n \in \mathcal{D}$  at time slot  $h$  is its proportional part of the total cost, i.e.,  $C_h(L(h), \delta(h)) \cdot \left( (l_n(h) + \delta_n(h)) / (L(h) + \sum_{n \in \mathcal{D}} \delta_n(h)) \right)$ , so that its cumulative extension for all time-slots is given by:

$$f_n(\mathbf{l}, \delta) \triangleq \sum_{h \in H} K_h \left( L(h) + \mathbf{1}^T \delta(h) \right) (l_n(h) + \delta_n(h)) + M(\delta_n) \quad (3)$$

where  $\mathbf{l}_n$  is a convex function;  $\mathbf{l}_n \in \Omega_{\mathbf{l}_n}$ ; all regions  $\Omega_{\mathbf{l}_n}$  are convex, compact and independent among users; and  $\mathbf{1}$  is a column vector of appropriate size. Additionally, we have included a penalization term  $M(\delta_n)$  that includes management or distribution costs, which accounts for the extra expenses that the supply authority has to account for during the pricing process. We separate these penalization terms from the energy prices due to the inherent unpredictable nature of the energy demand, which is an additional consideration with regards to the models in references [2], [10].

Since we consider an arbitrary form of  $\mathbf{l}_n$  and  $\Omega_{\mathbf{l}_n}$ , our robust proposal can be adapted to different demand-side models. For instance, in the model from [2],  $\mathbf{l}_n$  has a linear expression which is a function of the generation and storing capabilities of its users. Likewise, the model from [10] optimizes an energy user demand scheduling algorithm where  $\mathbf{l}_n$  is linear and the feasible region is convex. We can, therefore, adapt a robust energy price that accounts for real-time uncertainties. And apart from the models [2] and [10], we can trivially assume a third one: one in which all users are passive, have no elastic demand at all, and simply do not optimize their energy load profiles. In such case, our robust model can still be applied to determine worst-case estimation errors.

Finally, the error terms we have accounted for in (2) contribute to the user's monetary expense  $f_n(\mathbf{l}, \delta)$ , since they affect the real-time energy price. In order to deal with these unknown demand terms, in Section III we propose a worst-case analysis by solving a coupled min-max game. This analysis will allow the supply authority to provide for excess energy demands and establish a robust energy price that accounts for unplanned demand. In the next subsection we present how to integrate these worst-case error terms in a real-time scenario to obtain a robust model.

### C. Real-time Pricing Model

Once the robust price has been established one day ahead, if their demand remains within the worst-case estimation error limits, users are charged the amount of energy they demand in real time at the robust price. Outside of these limits, a penalty function is introduced to cover any extra expenses. Specifically, our proposed real-time cost model is

$$f_n^{\text{rt}}(\mathbf{l}, \delta^*, \mathbf{l}^{\text{rt}}) = \sum_{h \in H} K_h \hat{L}_r(h) (l_n^{\text{rt}}(h) + \Psi_{nh}(\mathbf{l}, \delta^*, \mathbf{l}^{\text{rt}})) + M(\delta_n) \quad (4)$$

where  $\mathbf{l}$  is the energy load profile obtained from the day-ahead optimization process,  $\delta^*$  is the worst-case estimation



error (which is analyzed in Section III), and

$$\hat{L}_r(h) = \sum_{m \in \mathcal{D}} l_m(h) + \delta_m^*(h) \quad (5)$$

is the robust total demanded energy plus worst-case estimation errors, calculated in the day-ahead optimization process. Variable  $l^r$  refers to the actual energy load that users demand in real-time, and  $\Psi_h$  is a penalty function that increases the cost whenever a user deviates from its specified load profile range; for example the following function is a real-time cost model inspired by [4]:

$$\begin{aligned} \Psi_{nh}(l, \delta^*, l^r) = & \nu_h(l_n(h) - \delta_n^*(h) - l_n^r(h))^+ \\ & + v_h(l_n^r(h) - l_n(h) - \delta_n^*(h))^+ \end{aligned} \quad (6)$$

where  $\nu_h$  and  $v_h$  are scalar terms and  $(\cdot)^+ = \max(\cdot, 0)$ . The penalty values discourage the return of unused energy during night or low demand times, and higher use of energy during day or high demand times.

Note that the penalty function proposed in (6) differs from [4] in that it does not charge any extra cost within the uncertainty margins. The reason is that our robust algorithm already takes into account the extra costs of unplanned demand from the production model. Note also that a user can still deviate further than the worst-case error estimates and, in such case, some penalty function should dissuade the user from these extremes; nevertheless, these deviations are limited with some probabilistic confidence.

Thus, our robust model acknowledges the cost of producing a certain amount of energy plus uncertainties, whereas current real-time models (like [4]) only establish a surcharge for extra demand. Robust pricing models are of interest for energy suppliers as they take into account production costs under real-time price fluctuations. This is an important remark, because the real-time demand affects the whole energy price, and not only the extra demanded energy.

#### D. Day-ahead Optimization Process

We can now introduce the game  $\mathcal{G}_\delta = \langle \Omega_l, f \rangle$  that minimizes the user's monetary expense with fixed  $\delta$ :

$$\mathcal{G}_\delta : \begin{cases} \min_{l_n} & f_n(l_n, l_{-n}, \delta) \\ \text{s.t.} & l_n \in \Omega_{l_n}, \end{cases} \quad \forall n \in \mathcal{D}$$

where  $l_{-n} = [l_1, \dots, l_{l-1}, l_{l+1}, \dots, l_{|\mathcal{D}|}]$  represents the rest of the other user's policies, which influence the user's decision. We consider the Nash Equilibrium (NE) as the solution concept of interest for this game, defined as a feasible point  $l^* = [l_1^*, \dots, l_{|\mathcal{D}|}^*]$ , such that it satisfies

$$f_n(l_n^*, l_{-n}^*, \delta) \leq f_n(l_n, l_{-n}^*, \delta) \quad \forall l_n \in \Omega_{l_n} \quad (7)$$

for every player  $n \in \mathcal{D}$ . In our framework,  $\mathcal{G}_\delta$  has at least an equilibrium point, which can be reached through a convergent algorithm as established by the following theorem:

**Theorem 1.** *The following holds for the game  $\mathcal{G}_\delta$  with given and fixed error values  $\delta$ :*

- a) *It has a nonempty, compact and convex solution set.*
- b) *The individual cost values for the payoff functions of each player are equal for any NE solution of the game.*

- c) *The game is monotone and NE solutions can be reached through algorithms of the kind proposed in [11].*

*Proof:* See Appendix A. ■

Convergent algorithms to reach such NE solutions are described in [11], which depend specifically on the convex structure of the game. In particular, they require a single-loop best response algorithm if the game is strongly monotone, or a double-loop best-response algorithm if it is just monotone (with an added proximity term). The specific steps, which are similar to those described in [2, Alg.1] for their particular model, but including fixed error terms, are as follows. First, users determine their respective  $l_n$  based on a day-ahead optimization process (considering  $\delta$  fixed), where every user knows their energy consumption requirements and energy costs, and calculate their best strategies depending on the energy prices. Then, all users communicate their intended energy load profile  $l_n$  to the central unit and energy prices are recalculated. This procedure is repeated iteratively until all users converge to an NE.

### III. WORST-CASE ANALYSIS OF THE ENERGY LOAD PROFILES

In the formulation of  $\mathcal{G}_\delta$  we implicitly assumed that  $\delta$  is given or known a priori, or otherwise is set to zero. However, these error profiles are in fact unknown random terms that we added in order to correct any deviation from the real performance. In this section, these error terms will be treated as optimization variables, and analyzed from a worst-case performance perspective. This allows the regulation authority to jointly consider both the price and management/availability costs of energy. With this idea in mind, it is realistic to consider bounded error terms, represented by

$$\Delta_h \triangleq \{\delta(h) \in \mathbb{R}^{|\mathcal{D}|} \mid \|\delta(h)\|_2^2 \leq \alpha(h)\} \quad \forall h \in H \quad (8)$$

where  $\alpha(h)$  are prefixed in advance and depend on the time-slot  $h$  due to more or less confidence in the regulator predictions. In practice, these values can be inferred by the regulation authority, based on historic data from consumers and expected variations. Quadratic constraints on the error terms are common in the literature, see e.g. [3], [12].

In order to propose this error model, we have assumed two conditions that are satisfied generally in practice: *i)* independence of the error terms among users, and *ii)* no knowledge of the specific probabilistic error profiles of users. Property *i)* is satisfied if we assume that users alter their predicted energy profiles independently (e.g. a user decides to charge an electric vehicle for an unexpected event). The energy profiles of different users can be correlated, but what we assume to be independent are the deviation errors from these users. Property *ii)* implies that if we do not have any knowledge of the error distribution of a specific user, then it is sensible to substitute such profile with the error distribution of an average user profile. By the central limit theorem such average user profile will approximate a Gaussian distribution. Then, the error contribution of all users can be modeled as a Gaussian distribution of zero mean, and variance the sum of all user

variances. Specifically, equation (8) represents an upper bound to the overall network error variance.

In the following, we analyze a scenario that considers that every player has to account individually for their worst-case error terms and is charged accordingly. The model assumes that each player is responsible for some maximum error profiles in joint consideration with the other players.

Before the analysis, we bring up a useful result that guarantees global optimality for a maximization problem with quadratic convex objectives and constraints.

**Theorem 2.** (from [13, Th.3]). *Given the problem*

$$\begin{aligned} \max_{x \in \mathbb{R}^n} \quad & \frac{1}{2}x^T A x + b^T x + c \\ \text{s.t.} \quad & \frac{1}{2}x^T Q x + d x + e \leq 0 \end{aligned} \quad (9)$$

where  $A \neq 0, Q \in \mathbb{R}^{n \times n}$  are positive semidefinite matrices,  $b, d \in \mathbb{R}^n$  and  $c, e \in \mathbb{R}$ , and Slater's condition is satisfied: then  $\bar{x}$  is a global maximizer iff there exists  $\bar{\lambda} \geq 0$  satisfying

$$A\bar{x} + b = \bar{\lambda}(Q\bar{x} + d) \quad (10)$$

$$-A + \bar{\lambda}Q \text{ is positive semidefnite.} \quad (11)$$

Note that (10) is a KKT necessary condition, plus a second requirement (11) that will guarantee global maximum.

#### A. Worst-Case Min-max Game Formulation

In this section we consider a worst-case situation where the error terms are coupled among all of the users. We analyze which are the user's best response strategies and how to calculate the worst-case error terms. In particular, this transforms  $\mathcal{G}_\delta$  into a game where all users have to solve a min-max cost function, with coupled interactions and constraints. The new game formulation  $\mathcal{G}_m$ , where each user  $n$  has to determine variables  $\mathbf{l}_n$  and  $\delta_n$ , is given by

$$\mathcal{G}_m : \begin{cases} \min_{\mathbf{l}_n} \max_{\delta_n} f_n(\mathbf{l}_n, \mathbf{l}_{-n}, \delta_n, \delta_{-n}) \\ \text{s.t.} \quad \delta(h) \in \Delta_h, \quad \forall h \in H \\ \mathbf{l}_n \in \Omega_{\mathbf{l}_n} \end{cases} \quad \forall n \in \mathcal{D},$$

where region  $\Omega_{\mathbf{l}_n}$  is convex, compact and nonempty. The min-max of game  $\mathcal{G}_m$  for every user  $n \in \mathcal{D}$  represents a game with a convex objective function in variables  $\mathbf{l}_n$  and convex regions. Note that for the specific case that  $M(\delta_n)$  is quadratic and convex in variable  $\delta_n$ , then the maximization problem falls into the category to readily use Theorem 2, since all equations are quadratic. More generally, we can provide the following theorem for any continuous function  $M(\delta_n)$ :

**Theorem 3.** *The following holds for game  $\mathcal{G}_m$  with min-max objectives:*

- It has a nonempty, compact and convex solution set.*
- It is monotone in variables  $\mathbf{l}_n$ , and NE can be reached through algorithms of the kind proposed in [11].*

*Proof:* The max operation inside the objective of each user in game  $\mathcal{G}_m$  is performed over a convex and compact set and, therefore, it is well defined (it always has a solution). The maximum function in  $\mathcal{G}_m$  preserves convexity of the objective in variable  $\mathbf{l}_n$ . Furthermore, if the point that maximizes the

objective is unique, then the objective is differentiable in  $\mathbf{l}_n$  with gradient  $\nabla_{\mathbf{l}_n} f_n(\mathbf{l}_n, \mathbf{l}_{-n}, \delta_n^*, \delta_{-n})$ , and where  $\delta_n^* = \arg \max_{\delta_n} f_n(\mathbf{l}, \delta_n, \delta_{-n})$  denotes the maximal point. This property is guaranteed by Danskin's Theorem [14, Sec. B.5]. With this analysis in mind, the required assumptions from [11, Th.4.1.a] are satisfied and an NE exists. Furthermore, the Jacobian analysis performed for Theorem 1 (see Appendix A) remains the same in  $\mathcal{G}_m$ , with the exception of substituting the fixed  $\delta_n$  with the optimal solution of the max problem  $\delta_n^*$ . Likewise, we can claim that the game is monotone and, therefore, that the solution set is also convex [11, Th.4.1.b]. By Danskin's theorem, if the maximum is not a unique point then the gradient would become a subdifferential, and slight variations would have to be considered that include the use of multifunctions [15]. ■

Game  $\mathcal{G}_m$  falls into a rather novel category of min-max problems recently analyzed in [16]. Such reference proposes an alternative formulation for min-max games, where extra players are added to solve the maximization parts to decouple the min-max form. Moreover, reference [16] proves equivalence of NE for both formulations, and proposes solutions under monotonicity and other requirements. However, our problem  $\mathcal{G}_m$  falls into a more general form with individual max objectives (rather than a common one), where the monotonicity property does not hold. Therefore, the approach from [16] becomes unsuitable for distributed algorithms. Furthermore,  $\mathcal{G}_m$  incorporates global constraints which are not considered in the previous reference. For this reason, we have to develop a novel approach for this particular problem.

In order to analyze and solve the coupled maximization problems for all players, we first formulate an equivalent game  $\tilde{\mathcal{G}}_m(\boldsymbol{\lambda})$ , which is more tractable, together with a nonlinear complementarity problem (NCP) indicated by (12). The NCP is required to establish the equivalence together with other assumptions on the dual variables  $\lambda_h$ . We have also particularized the cost term to  $M(\delta_n) = \beta_m \|\delta_n\|_2^2$  with  $\beta_m \geq 0$ , which is motivated to penalize the user's local deviations. Furthermore, this particular choice will allow us to obtain tractable results for the game and the NCP:

$$\begin{aligned} \tilde{\mathcal{G}}_m(\boldsymbol{\lambda}) : \begin{cases} \min_{\mathbf{l}_n} \max_{\delta_n} p_n(\mathbf{l}_n, \mathbf{l}_{-n}, \delta_n, \delta_{-n}, \boldsymbol{\lambda}) \\ \text{s.t.} \quad \mathbf{l}_n \in \Omega_{\mathbf{l}_n} \end{cases} \quad \forall n \in \mathcal{D} \\ 0 < K_h + \beta_m \leq \lambda_h \perp -\left(\|\delta(h)\|_2^2 - \alpha(h)\right) \geq 0, \quad \forall h \end{aligned} \quad (12)$$

where  $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_{|H|}]^T$ ,  $a \perp b$  indicates  $a^T b = 0$ , and

$$\begin{aligned} p_n(\mathbf{l}_n, \mathbf{l}_{-n}, \delta_n, \delta_{-n}, \boldsymbol{\lambda}) = & \sum_{h \in H} (\beta_m - \lambda_h) \delta_n(h)^2 \\ & + \sum_{h \in H} K_h \left( L(h) + \sum_{k \in \mathcal{D}} \delta_k(h) \right) (\mathbf{l}_n(h) + \delta_n(h)). \end{aligned} \quad (13)$$

Next lemma shows the equivalence between  $\mathcal{G}_m$  and  $\tilde{\mathcal{G}}_m(\boldsymbol{\lambda})$ .

**Lemma 1.** *Games  $\mathcal{G}_m$  and  $\tilde{\mathcal{G}}_m(\boldsymbol{\lambda})$  together with the NCP specified by equation (12) present the same NE solutions.*

*Proof:* The KKT system of equations of  $\mathcal{G}_m$  and those of game  $\tilde{\mathcal{G}}_m(\boldsymbol{\lambda})$  jointly with the NCP are the same, and therefore, any NE candidate solution can be derived from

either formulation. A maximum point in both problems exists since all regions are compact. Inequality  $\lambda_h \geq K_h + \beta_m$  guarantees that such point is maximum because Theorem 2 is satisfied for every user in  $\mathcal{G}_m$  and, likewise, is maximum in  $\tilde{\mathcal{G}}_m(\lambda)$  because it makes the objective concave in  $\delta_n$ . Therefore, the NE coincide. ■

Note that, in this equivalent formulation of  $\tilde{\mathcal{G}}_m(\lambda)$  we have focused on the special case where all  $\lambda_h$  are equal for all players for every  $h \in H$ . The equilibrium points of this formulation are normally referred to as “variational” or “normalized” solutions [17], and the practical implication is that they establish a common price on a resource for all agents. These solutions also retain some stability properties compared to other equilibrium points where the dual variables are different among users, see [17] for further details.

One benefit from dealing with the new formulation  $\tilde{\mathcal{G}}_m(\lambda)$  is that the *strong max-min property* is directly fulfilled if the objective is *convex-concave*, as opposed to  $\mathcal{G}_m$  where it remains *convex-convex* (see [18, Ex.3.14b] for further details). This implies that the min-max can be changed into a max-min and the solution is not altered, i.e., equivalence of both problems holds. We see this is true as long as  $\lambda_h \geq K_h + \beta_m$  because the objective is then concave in  $\delta_n$ . This allows us to analyze the monotonicity of the new game setting  $\tilde{\mathcal{G}}_m(\lambda)$  in these variables. Let us present a lemma that guarantees the existence of common  $\lambda_h$ 's satisfying the previous requirements:

**Lemma 2.** *There always exists  $\lambda_h$  satisfying (12) (i.e., making  $\tilde{\mathcal{G}}_m(\lambda)$  concave in  $\delta_n$ ) that is dual optimal in  $\mathcal{G}_m$ . A lower bound to this value is:  $\lambda_h > \beta_m + K_h$  for all  $h \in H$ .*

*Proof:* See Appendix B. ■

Now, we can establish the following result.

**Theorem 4.** *The following holds:*

- a) *Game  $\tilde{\mathcal{G}}_m(\lambda)$  has a nonempty and compact solution set.*
- b) *A variational solution of game  $\mathcal{G}_m$  always exists.*
- c) *If additionally  $\lambda_h > \frac{1}{2}K_h(|D| + 1) + \beta_m$ , then the game  $\tilde{\mathcal{G}}_m(\lambda)$  is strongly monotone in variables  $\delta_n$  for all  $n \in \mathcal{D}$ , it has a unique NE solution, and it can be reached by a distributed asynchronous best response algorithm of the kind proposed by [11, Alg.4.1].*

*Proof:* See Appendix B. ■

In order to solve  $\tilde{\mathcal{G}}_m(\lambda)$  and the NCP described by (12), we need to study the inner game formed from the coupled maximization problems in  $\delta$ . In order to do that, we assume that  $l$  is fixed. Specifically, we analyze the following game:

$$\left\{ \max_{\delta_n \in \mathbb{R}^{|H|}} p_n(l, \delta_n, \delta_{-n}, \lambda) \quad \forall n \in \mathcal{D} \right. \quad (14)$$

where (14) is solved jointly with the NCP (12). First, note that all users' objectives can be separated in  $|H|$  independent problems. This is possible because the objective function  $p_n$  is expressed as a sum over  $h \in H$  and (12) is expressed separately for every  $h \in H$ .

Each individual problem has a quadratic form and, hence, Theorem 2 can be applied to each of them. We can transform the model given by (12) and (14) into a form resembling (9), and identify terms  $A = K_h + \beta_m$  and  $Q = 1$ . We conclude

that (11) is satisfied if  $\lambda_h \geq K_h + \beta_m$ . Finally, we obtain the first order necessary conditions from (14) for all  $n \in \mathcal{D}$ :

$$K_h \left( l_n(h) + L(h) + \sum_{k \neq n} \delta_k(h) \right) + 2 \left( K_h + \beta_m - \lambda_h \right) \delta_n(h) = 0,$$

and form a system of equations together with (12). By separating variables and solving, we get

$$\delta_n(h) = \frac{\sqrt{\alpha(h)}(l_n(h) + L(h) + \sum_{k \neq n} \delta_k(h))}{\sqrt{\sum_n (l_n(h) + L(h) + \sum_{k \neq n} \delta_k(h))^2}}, \forall n \in \mathcal{D} \quad (15)$$

$$\lambda_h = K_h + \beta_m + K_h \frac{\sqrt{\sum_n (l_n(h) + L(h) + \sum_{k \neq n} \delta_k(h))^2}}{2\sqrt{\alpha(h)}} \quad (16)$$

where we have chosen positive sign to satisfy  $\lambda_h \geq K_h + \beta_m$ .

Since the solution to the fixed point equations is derived from conditions (10)–(11), they are, therefore, global maximum of the individual objective functions. As no player can further maximize their objectives given the other players strategies, they form an NE. If the game is strongly monotone (Th. 4c is satisfied), then asynchronous fixed-point iterations over these equations will converge to the global maximum. If the game is not strongly monotone, then the set of optimal equations given by (15)–(16) have to be solved in an alternative fashion. We propose a fixed-point iterative mapping that results from calculating equation (15) repeatedly, plus a projection operation. The mapping in vectorial form becomes:

$$T_h(\delta(h), a_h) = \Pi_{X_h} \left( \sqrt{\alpha(h)} \frac{a_h + A\delta(h)}{\|a_h + A\delta(h)\|} \right) \quad (17)$$

where  $a_h = L(h) + l(h)$ ,  $A = \mathbf{1}\mathbf{1}^T - \mathbf{I}$  with size  $|\mathcal{D}| \times |\mathcal{D}|$  and  $\Pi_{X_h}(\cdot)$  represents the euclidean projection onto region  $X_h$ . Such region  $X_h$  is defined as

$$X_h \triangleq \left\{ \delta(h) \in \mathbb{R}_+^{|\mathcal{D}|} \mid \mathbf{1}^T \delta(h) + \frac{1}{|\mathcal{D}|-1} \mathbf{1}^T a_h - \sqrt{\alpha(h)|\mathcal{D}|} \geq 0 \right\} \quad (18)$$

where  $\mathbb{R}_+^{|\mathcal{D}|}$  refers to the nonnegative quadrant, so that  $X_h$  corresponds to the upper halfspace region limited by the hyperplane in the definition.

Next we give the following properties of the mapping  $T_h$ :

**Theorem 5.** *Given the mapping  $T_h(\delta(h), a_h)$  defined in (17) where  $a_h$  is fixed, then it follows:*

- a)  *$T_h : X_h \rightarrow X_h$  is a self-map for every  $h \in H$ .*
- b)  *$T_h$  is a contraction mapping.*
- c) *The update rule  $\delta(h)^{k+1} = T_h(\delta(h)^k, a_h)$  converges to a unique point  $\delta(h)^* \in X_h$  that solves the fixed point equation:*

$$\delta^*(h) = T_h(\delta^*(h), a_h).$$

- d) *The point  $\delta^*(h)$  is an NE of game  $\tilde{\mathcal{G}}_m(\lambda)$ , with  $\lambda$  satisfying (12) and fixed strategies  $l(h)$ .*

*Proof:* See Appendix C. ■

Because of Theorem 5 and the equivalence principle of Lemma 1, the fixed-point solution of the mapping also maximizes the objectives of game  $\mathcal{G}_m$ . Thus, the mapping provides an iterative method to find a solution of the game, regardless of its monotonicity properties.

Region  $X_h$  in (18) becomes a technical adjustment that allows the operator to be a contraction self-mapping. It has an



involved interpretation, but in simple terms it limits the search space of the fixed-point to the upper region of the hyperplane. In particular, in such upper halfspace the mapping is always a contraction, while in the lower halfspace such property is not always satisfied. Therefore, a fixed point in  $X_h$  always exists, and a recursive algorithm will converge to such point.

Finally, we can explicitly give an expression of the projection operation into the halfspace  $X_h$ , which is given by

$$\Pi_{X_h}(x) \triangleq \begin{cases} x & \text{if } x \in X_h \\ x + \left( \sqrt{\alpha(h)} - (A^{-1}\mathbf{a}_h + x)^T \frac{\mathbf{1}}{\|\mathbf{1}\|} \right) \frac{\mathbf{1}}{\|\mathbf{1}\|} & \text{if } x \notin X_h, \end{cases}$$

which only involves linear operations.

### B. Distributed Robust Algorithm

Distributed implementations are beneficial for the smart grid because of their computation scalability. Therefore, we propose the distributed Algorithm 1.

The algorithm solves  $\mathcal{G}_m$ , and because of the monotonicity shown in Theorem 3 in variable  $\mathbf{l}$ , convergence is guaranteed. The proximal term included in step (S.3) is necessary if  $\mathcal{G}_m$  is only monotone, and  $\tau$  needs to be sufficiently large and satisfy that the game becomes strongly monotone, as indicated in [11, Cor.4.1]. Any optimization toolbox would suffice to solve the strongly convex problem indicated in (S.3), which has a unique solution. In step (S.4.), we solve the inner max game  $\tilde{\mathcal{G}}_m(\boldsymbol{\lambda})$  among all players with fixed components  $\mathbf{l}_n^{i+1}$ . In order to do so, we iteratively calculate  $T(\delta(h)^k, \mathbf{a}_h)$  until convergence, which is guaranteed by Theorem 5. The main outer loop is also repeated until convergence.

Regarding the information exchange costs, users solve their local optimization problems, and then communicate their intended energy load profiles to the central unit. The central unit aggregates all energy loads and computes the individual worst-case error terms and the robust energy price defined in (5). Then, the robust energy price and the individual worst-case error estimates are communicated back to the users (privacy of individual values is preserved). Finally, the users re-optimize their new energy load profiles.

It seems sensible that the inner loop described on step 4 of Algorithm 1 is performed by the central unit. It involves simple operations described by the mapping  $T_h(\delta(h), \mathbf{a}_h)$ , which scales linearly with the number of users  $|\mathcal{D}|$ , and requires no communication in order to find a solution. An alternative computation of the error terms in a distributed fashion is also possible, but it would increase the required communication.

## IV. SIMULATION RESULTS

In this section, we present simulation results from our robust analysis. First, in Subsection IV-A we present a real-time cost comparison with a non-robust model to illustrate that the robust model can effectively reduce the monetary costs. Secondly, in Subsection IV-B, we compare our algorithm with a model formed by naive users who do not take into account the worst-case error terms. We show that the monetary losses in this case are not negligible, and emphasize the importance of optimizing the energy loads taking into account the error terms. In addition, we include convergence results.

---

### Algorithm 1 Distributed robust algorithm for game $\mathcal{G}_m$

---

(S.1) Given  $K_h$  for all  $h \in H$ . Set iteration index  $i = 0$ .

Set  $\tau > 0$ . Initialize  $\delta^0(h) = \frac{\sqrt{\alpha(h)}}{\sqrt{|\mathcal{D}|}}$ .

(S.2) If a termination criterion is satisfied, STOP.

(S.3) Solve for every  $n \in \mathcal{N}$

$$\mathbf{l}_n^{i+1} = \arg \min_{\mathbf{l}_n \in \Omega_{\mathbf{l}_n}} f(\mathbf{l}_n, \mathbf{l}_{-n}^i, \delta^i) + \frac{\tau}{2} (\mathbf{l}_n - \bar{\mathbf{l}}_n^i)^2$$

(S.4) Iterate from  $k = 0$  until convergence

$$\hat{\delta}^{k+1}(h) = T(\hat{\delta}^k(h), \mathbf{a}_h^{i+1}), \quad \forall h \in H \quad (19)$$

where  $\mathbf{a}_h^{i+1} = \sum_{n \in \mathcal{D}} \mathbf{l}_n^{i+1}(h) + \mathbf{l}^{i+1}(h)$ .

Set the fixed-point solution:  $\delta^{i+1} \leftarrow \hat{\delta}^{k+1}$ .

(S.5) If an NE has been reached, update centroids  $\bar{\mathbf{l}}_n^i = \mathbf{l}_n^i$ .

(S.6) Set  $i \leftarrow i + 1$ . Go to (S.2).

---

In the following simulations, we used Algorithm 1 to determine  $\delta_n$  and  $\mathbf{l}_n$  applied to a scenario similar to the one proposed in [2]. In this kind of scenario, users decide upon local energy generation and storage strategies. The total number of users  $|\mathcal{D}|$  will vary between simulations, but the proportion of active users remains fixed in all of them, where  $|\mathcal{N}| = \lfloor |\mathcal{D}|/2 \rfloor$ . From the active users set  $\mathcal{N}$ , the number of users that generate and store energy is  $GS = |\mathcal{N}| - 2\lfloor |\mathcal{N}|/3 \rfloor$ , the number of users who only generate is  $G = \lfloor |\mathcal{N}|/3 \rfloor$  and the number of users who only store energy is  $S = \lfloor |\mathcal{N}|/3 \rfloor$ . The day is divided in  $|H| = 24$  time slots. The average consumption per user is of 4.5 kWh with cost coefficients  $K_{\text{day}} = 1.5K_{\text{night}}$ , where we assumed there is lower price at times  $h_{\text{night}} = [1, \dots, |H|/3]$  and higher price at time slots  $h_{\text{day}} = [|H|/3 + 1, \dots, |H|]$ . These values are determined so that the initial average price for the day without any demand-side management is of 0.1412 £/kWh (same as in [2]).

Other parameters involving active users with generation capabilities,  $GS$  and  $G$ , are:  $g_n^{(\max)} = 0.4$  kWh/h, which is the maximum energy production capability per hour by the user's owned infrastructure; and  $\gamma_n^{(\max)} = 0.8g_n^{(\max)}$ , which is the maximum energy production capability per day. For active users owning storing devices,  $GS$  and  $S$ , we define the following variables: the hourly batteries leakage rate  $\alpha_n = \sqrt[24]{0.9}$ , the charge efficiency  $v_n^+ = 0.9$ , the efficiency at discharge  $v_n^- = 1.1$ , the capacity of the storage device  $c_n = 4$  kWh, the maximum charge rate per hour  $s_n^{(\max)} = 0.125c_n$ , the charge level at the beginning of the day  $q_n(0) = 0.25c_n$  and the difference of charge at the next day  $\epsilon_n = 0$ . All these variables are further explained in reference [2]. In essence, the parameters presented form a convex model of the problem that can be solved using convex optimization solvers.

The energy demand profiles per user have been simulated using a typical day profile similar to the one used in [2] plus some random Gaussian noise that adds some variance to the users. The  $\alpha(h)$  terms have been fixed to be the 10% of the total energy demand per time slot  $h \in H$ .

#### A. Real-time Cost Comparison

In this subsection, we present a comparison of our robust model with the non-robust model from [4], when users de-

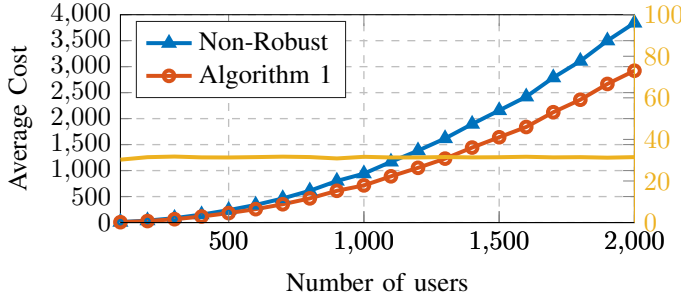


Fig. 1: Real time averaged cost functions over 100 simulations comparing robust function (4) and (20). The left axis shows the monetary cost of all users and the right axis shows (in %) the mean relative gain when using the robust algorithm.

viate from their day-ahead optimized energy load profiles. Specifically, we present the average cost function over 100 independent simulations, where the real time error deviations are given by white Gaussian processes with zero mean and variance  $\alpha(h)/|D|$  for each user, and where the number of users  $|D|$  ranges from 100 to 2000. The cost function we use to evaluate the performance in real-time of Algorithm 1 is described in (4). Likewise, the cost function for user  $n$  we compare with, denoted as ‘non-robust’, has the following form:

$$f_n^{\text{nr}}(\mathbf{l}, \delta^*, \mathbf{l}^{\text{rt}}) = \sum_{h \in H} K_h \hat{L}_{wc}(h) \left( l_n^{\text{rt}}(h) + \nu_h (l_n(h) - l_n^{\text{rt}}(h))^+ + v_h (l_n^{\text{rt}}(h) - l_n(h))^+ \right), \quad (20)$$

which has no error terms in contrast to (4). Note also that we are representing the average cost of all users in the network. Finally, values  $\nu_h$  and  $v_h$  are set to  $\nu_h = 0.2\kappa$  for  $h \in [0, \dots, 8]$ ,  $\nu_h = 0.8\kappa$  for  $h \in [9, \dots, 24]$  and  $v_h = (\kappa - \nu_h)$ , where  $\kappa = \sqrt{|D|/\alpha(h)}$ . The specific simulation parameters  $\nu_h$  and  $v_h$  are obtained from [4], and parameter  $\kappa$  is introduced to incorporate the uncertainty of the real-time consumption in the penalty terms. Note that the value of  $\kappa$  is the inverse of the deviation of every user’s real-time error distribution.

The average results of the cost models are presented in Figure 1. The term  $M(\delta_n)$  is not present in (20), so we set  $M(\delta_n) = 0$  in (4) for a fair comparison. Commenting on these results, we observe that the cost for the users is significantly lower than with a penalty based model as (20). Therefore, this model effectively reduces the monetary expenses of all users in the real-time market, while at the same time provides a reliable production cost estimate to the energy supplier.

### B. Comparison of Algorithm 1 with naive users

In order to further support our results, we include the performance of our robust algorithm and compare it to a scenario where users do not consider the worst-case error terms in their energy load profiles. In Figure 2, we plot the monetary cost of all users in different sizes of the smart grid. The number of users in this plot varies from 100 to 2000 users, and the total number of time slots is  $|H| = 24$ . The curve we refer to as ‘naive users’ is the solution to the game without accounting for any error terms (i.e.,  $\delta = 0$ ), but with added worst-case calculations in the price determined by the energy

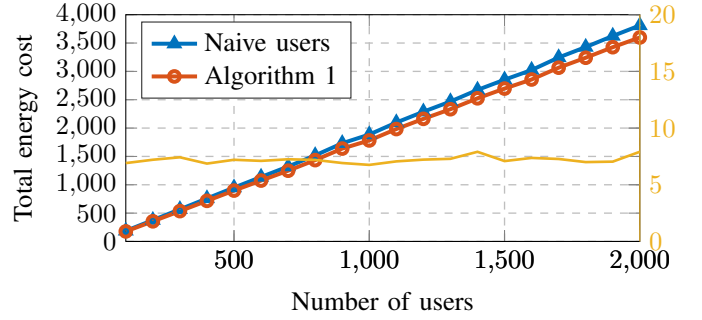


Fig. 2: Energy cost of all players vs. number of users. The left axis shows the monetary cost and the right axis shows the mean relative gain (in %) when using the robust algorithm.

supplier. On the other hand, the robust curve represents the total energy cost when using Algorithm 1, which does take into account the worst-case error terms. The average relative gain over all users when using the robust algorithm is around the 7%, which remains almost constant with the number of users; these savings values are represented in the right hand axis of the figure for different number of users. This result illustrates that users should adjust their energy load profiles after taking the worst-case error terms into account. For these simulations we fixed the term  $\beta_m = 0.001$ .

In Figure 3a, we plot the convergence rate of Algorithm 1, which depicts the averaged convergence rate particularized for 1000 users. The convergence rate is very similar to the results presented in earlier approaches (see, e.g., as [2, Alg.1]). Our algorithm has the same algorithmic steps except for the inner loop involved in step 4, which we didn’t observe to impact the computation time. In Figure 3b, we plot the convergence speed of the contraction mapping from step (S.4) in Algorithm 1. The x-axis shows the number of iterations and the y-axis shows the magnitude  $\sum_{h \in H} \|\delta^{k+1}(h) - \delta^k(h)\|$ . The stopping criteria was set to  $10^{-8}$ . Recall that convergence is guaranteed since the computation in (S.4) is a contraction mapping.

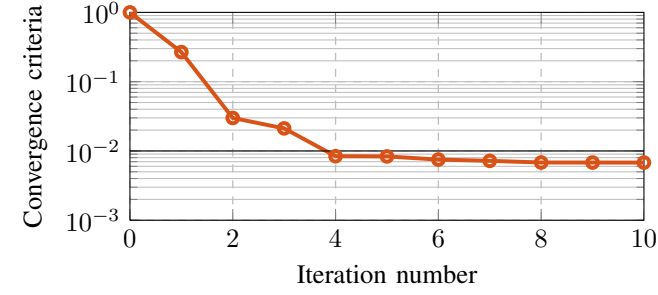
The proposed algorithm was implemented in Matlab R2015a using an Intel i5-2500 CPU @ 3.30GHz. The user’s optimization problems were run in sequential order (although in practice they can run in parallel). The local convex optimization problems have unique optimal points, so any optimization solver will produce similar results as ours. In particular, we used the *fmincon* function from Matlab. The whole simulation took about 61.38 seconds for 1000 users.

As a conclusion, these simulations show consistent results in which there is an actual benefit of around the 7% when using a robust algorithm vs. a naive one (i.e., that does not take into account the error terms for the optimization process), while the extra computation required is minimal.

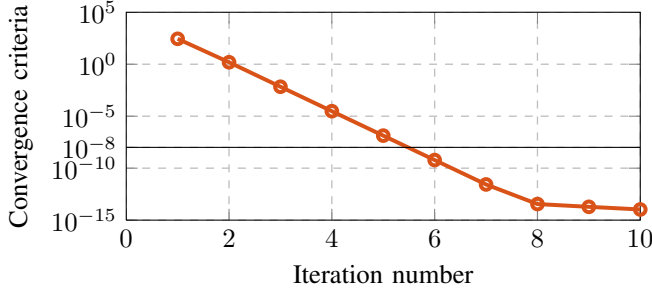
## V. CONCLUSION

We presented a demand-side management model that takes into account deviations in the estimated consumption profiles, and analyzed the worst-case effects of these terms. We considered a scenario where the error terms are calculated by the energy users (yielding a min-max game) which includes non-concave maximizing problems for all users. This method





(a) Convergence rate of Algorithm 1 with convergence criteria  $\|l^i - l^{i-1}\|_2 / \|l^i\| \leq 10^{-2}$ .



(b) Convergence rate of  $\delta^k(h)$  from step 4 in Algorithm 1 with convergence criteria  $\sum_{h \in H} \|\delta^{k+1}(h) - \delta^k(h)\| \leq 10^{-8}$ .

Fig. 3: Convergence rates for 1000 users.

provides estimates of the worst-case energy deviations and aggregate loads of the network, and further considers robust pricing as an effect of both global energy demand, as well as uncertain real-time demand costs of the supply authority. The problems can be solved globally due to the quadratic structure they present in the objective and constraints. With a novel analysis we are able to prove existence of NE, as well as to propose a novel convergent distributed algorithm to find such solutions. The proposed algorithm has very little computational complexity, and provides significant monetary savings (around the 7% in our simulations). We further illustrated the benefit of our robust model giving flexibility to the users within certain margins while providing a robust production price to the energy supplier.

#### APPENDIX A

*Proof of Theorem 1:* We first need to establish that function  $f_n(l_n, l_{-n}, \delta)$  is convex provided that  $l_n$  is a convex function. Indeed, this is satisfied because of the scalar composition rule [18, Sec.3.2.4], where  $f_n(l_n, l_{-n}, \delta)$  is monotone if  $L > 0$ , as required by (1). Then, because  $\Omega_{l_n}$  is convex by assumption, the individual user problem is also convex.

The rest of the results we establish are similar to the ones considered in [2, Th.1 & Th.2]. These are based on the equivalence of game  $\mathcal{G}_\delta$  to a variational inequality, as explained in [11]. To show a) is satisfied, it is sufficient to verify that the regions  $\Omega_{l_n}$  are compact and convex, which is satisfied by definition of  $\mathcal{G}_\delta$ , and that the objective functions of all players are differentiable and convex in variables  $l_n$ ,  $\forall n \in \mathcal{D}$ . Indeed, the convexity is guaranteed since the objective functions are quadratic with positive terms in the

squared elements so that the Hessian results into a diagonal matrix of the form  $\text{diag}(\{K_h\}_h)$  that is positive definite. Hence, the claim is satisfied applying [11, Th.4.1].

The justification of b) is readily available by observing that the objective is convex in  $l_n(h)$  and that other players actions only affect the individual objective through the aggregate  $L(h)$ . Therefore, the maximum value is unique for each player given any strategy profile  $l_{-n}^*$  since the maximum does not depend on individual decisions. This shows that even if there are multiple NEs that reach the maximum of  $f_n(l_n, l_{-n})$ , the objective value of the user remains constant.

The game is monotone if the Jacobian  $JF_{\mathcal{G}_\delta}(l, \delta)$  is positive semidefinite, where  $JF_{\mathcal{G}_\delta}(l, \delta) = (\nabla_{l_m} F_n(l, \delta))_{n,m \in \mathcal{D}}$ ,  $F_n = (\nabla_{l_n} f_n(l_n, l_{-n}, \delta))_{n \in \mathcal{D}}$  and where  $f_n(l_n, l_{-n}, \delta)$  is described in (3). The Hessian  $\nabla_{l_n, l_n}^2 f_n(l_n, l_{-n}, \delta) = \text{diag}(\{2K_h\}_{h \in H})$  is positive definite and the other terms  $\nabla_{l_m, l_n}^2 f_n(l_n, l_{-n}, \delta) = \text{diag}(\{K_h\}_{h \in H})$  for  $m \neq n$  are also positive definite and, therefore, so is  $JF_{\mathcal{G}_\delta}(l, \delta)$ . The claim is satisfied applying [11, Eq.(4.8)]. ■

#### APPENDIX B

*Proof of Lemma 2:* We use the *extreme value theorem* from *calculus* (also known as the Bolzano-Weierstrass theorem), which states that a real-valued function on a nonempty compact space is bounded above, and attains its supremum. Then, Theorem 2 states that if a maximum exists, there also exists a  $\bar{\lambda}$  that satisfies equation (11) and, therefore, we necessarily have that there exists some  $\lambda_h \geq K_h + \beta_m$  that solves the problem. ■

*Proof of Theorem 4:* To proof part a) of the theorem (the existence of NE in  $\tilde{\mathcal{G}}_m(\lambda)$ ) we use a result from [19, Th.1]. We need to show that assumptions A1-A3 and A5 are satisfied from the mentioned reference (A4 is not required). Indeed,  $\tilde{\mathcal{G}}_m(\lambda)$  has a number of finite players, the regions for all players are compact and convex, and all objective functions are continuous on these regions (which satisfy A1-A3). Condition A5 is satisfied if we show that  $p_n(l, \delta_n, \delta_{-n}, \lambda)$  is concave in  $\delta_n$ , which is true if its hessian is negative semidefinite. Indeed,  $\nabla_{\delta_n, \delta_n}^2 p_n = -2\text{diag}(\{\lambda_h - K_h - \beta_m\})$  is diagonal, and is negative semidefinite if  $\lambda_h \geq K_h + \beta_m$ . This also justifies that the *strong max-min property* of  $\tilde{\mathcal{G}}_m(\lambda)$  holds and the min-max can be transformed into a max-min without altering the result. This concludes the proof of part a) of the theorem.

To prove part b) we need Lemma 1, which guarantees the equivalence of  $\mathcal{G}_m$  and  $\tilde{\mathcal{G}}_m(\lambda)$  when (12) is satisfied. Since by Lemma 2 we have  $\lambda_h \geq \beta_m + K_h$  satisfied for all players, we conclude that a variational solution (a solution in which all  $\lambda_h$  are shared among all users) exists. Note that the game  $\mathcal{G}_m$  cannot satisfy condition A5 from previous paragraph directly and, therefore, the reformulation as  $\tilde{\mathcal{G}}_m(\lambda)$  is convenient to conclude the existence of an NE.

In order to prove part c), it suffices to show that the Jacobian  $JF_{\tilde{\mathcal{G}}_m(\lambda)}(\delta(h))$  of the variational inequality associated with the game is negative definite [11, (4.8)]. This is defined as  $JF_{\tilde{\mathcal{G}}_m(\lambda)}(l, \delta) = (\nabla_{l_m} F_n(l, \delta))_{n,m \in \mathcal{D}}$ , where  $F_n = (\nabla_{\delta_n} p_n(l, \delta_n, \delta_{-n}, \lambda))_{n \in \mathcal{D}}$  and  $p_n(l, \delta_n, \delta_{-n}, \lambda)$  is described in (13). The Hessian for each  $h \in H$  is diagonal

on all players, and the Jacobian on each  $h \in H$  becomes

$$\nabla_{\delta(h), \delta(h)}^2 p_n(\mathbf{l}, \boldsymbol{\delta}, \boldsymbol{\lambda}) = (K_h + 2\beta_m - 2\lambda_h)\mathbf{I} + K_h \mathbf{1}\mathbf{1}^T \quad (21)$$

where  $\mathbf{I}$  is the identity matrix of size  $|\mathcal{D}| \times |\mathcal{D}|$ , and  $\mathbf{1}$  is column vector of ones of length  $|\mathcal{D}|$ . The Hessian matrix becomes diagonally dominant with negative diagonal entries for  $\lambda_h > \frac{1}{2}K_h(|\mathcal{D}| + 1) + \beta_m$  and, thereby, negative definite [20, Cor.7.2.3]. If such condition is satisfied, then the game is strongly monotone and it has a unique NE that can be reached by a suitable algorithm [11]. ■

#### APPENDIX C

For notation simplicity, we refer in this section  $T_h(\delta(h), \mathbf{a}_h)$  defined in (17) simply as  $T(\mathbf{x}) = \Pi_X(\sqrt{\alpha} \frac{\mathbf{A}\mathbf{x} + \mathbf{a}}{\|\mathbf{A}\mathbf{x} + \mathbf{a}\|})$ , where  $\mathbf{a} = \mathbf{a}_h$  is of size  $|\mathcal{D}| \times 1$ ,  $\mathbf{A} = \mathbf{1}\mathbf{1}^T - \mathbf{I}$  of size  $|\mathcal{D}| \times |\mathcal{D}|$ ,  $x = \delta(h)$  and where we drop all indexes on  $h$ .

Before proving Theorem 5 we first introduce the following general result:

**Lemma 3.** *Given any two vectors  $x_1, x_2 \in \mathbb{R}^{|\mathcal{D}|}$ , the following holds:*

$$\left\| \frac{x_1}{\|x_1\|} - \frac{x_2}{\|x_2\|} \right\| \leq \left\| \frac{x_1}{\gamma_s} - \frac{x_2}{\gamma_s} \right\| \leq \left\| \frac{x_1}{\gamma} - \frac{x_2}{\gamma} \right\|$$

where  $\gamma_s \triangleq \min\{\|x_1\|, \|x_2\|\}$  and  $\gamma \triangleq \min_{x \in X} \|x\|$ , assuming in the last inequality that  $\gamma_s \neq 0 \neq \gamma$ .

*Proof:* We start with the first inequality and assume without loss of generality that  $\|x_1\| \geq \|x_2\|$ . Introduce the shorthands  $a = \frac{x_1}{\|x_1\|} - \frac{x_2}{\|x_2\|}$ ,  $b = \frac{x_1}{\|x_2\|} - \frac{x_1}{\|x_1\|}$  and, therefore,  $a + b = \frac{x_1}{\|x_2\|} - \frac{x_2}{\|x_2\|}$ . By the definition of scalar product,  $\|a + b\|^2 = \|a\|^2 + \|b\|^2 + 2a^T b$ . Hence, if  $a^T b \geq 0$ , we necessarily have  $\|a + b\| \geq \|a\|$ , since all the sums involve positive scalars. Then, the inner product results:

$$a^T b = \left( \frac{x_1}{\|x_1\|} - \frac{x_2}{\|x_2\|} \right)^T \left( \frac{x_1}{\|x_2\|} - \frac{x_1}{\|x_1\|} \right) \quad (22)$$

$$= \frac{\|x_1\|}{\|x_2\|} - 1 - \frac{x_2^T x_1}{\|x_2\|^2} + \frac{x_2^T x_1}{\|x_1\| \|x_2\|} \quad (23)$$

$$= \left( \frac{\|x_1\|}{\|x_2\|} - 1 \right) (1 - \cos \theta) \geq 0 \quad (24)$$

where we have used  $\cos \theta = \frac{x_2^T x_1}{\|x_1\| \|x_2\|}$ .

The proof of the second inequality is more immediate, since  $\frac{\gamma_s}{\gamma} \geq 1$  we have  $\|a + b\| \leq \left\| \frac{x_1}{\gamma} - \frac{x_2}{\gamma} \right\|$ . A graphical interpretation of this lemma is presented in Figure 4. ■

Now we use Lemma 3 to prove Theorem 5.

*Proof of Theorem 5:* It is clear by the projection operator  $\Pi_{X_h}(\cdot)$  that  $T(\mathbf{x})$  is a self-map onto itself, so *a)* is immediate.

To prove part *b)* we need to show that  $T(\mathbf{x})$  is a contraction:

$$\|T(\mathbf{x}_1) - T(\mathbf{x}_2)\| \leq q \|\mathbf{x}_1 - \mathbf{x}_2\| \quad (25)$$

for all  $\mathbf{x}_1, \mathbf{x}_2 \in X$ , and some  $q \in (0, 1)$ . We have

$$\begin{aligned} \left\| \sqrt{\alpha} \frac{\mathbf{a} + \mathbf{A}\mathbf{x}_1}{\|\mathbf{a} + \mathbf{A}\mathbf{x}_1\|} - \sqrt{\alpha} \frac{\mathbf{a} + \mathbf{A}\mathbf{x}_2}{\|\mathbf{a} + \mathbf{A}\mathbf{x}_2\|} \right\| &\stackrel{(i)}{\leq} \sqrt{\alpha} \left\| \frac{\mathbf{a} + \mathbf{A}\mathbf{x}_1}{\gamma} - \frac{\mathbf{a} + \mathbf{A}\mathbf{x}_2}{\gamma} \right\| \\ &\leq \frac{\sqrt{\alpha}}{\gamma} \|A\| \|\mathbf{x}_1 - \mathbf{x}_2\| = \frac{\sqrt{\alpha}}{\gamma} (|\mathcal{D}| - 1) \|\mathbf{x}_1 - \mathbf{x}_2\| \end{aligned}$$

where in step (i) we have applied the result from Lemma 3 and defined  $\gamma = \min_{\mathbf{x} \in X} \|\mathbf{a} + \mathbf{A}\mathbf{x}\|$ .

At this point, we can infer that (25) is satisfied with  $q = \frac{\sqrt{\alpha}}{\gamma} (|\mathcal{D}| - 1)$  and hence,  $T(\mathbf{x})$  is a contraction mapping in region  $X$  if  $\frac{\sqrt{\alpha}}{\gamma} (|\mathcal{D}| - 1) < 1$  is satisfied. In order to prove this last condition, we analyze the quadratic form  $\|\mathbf{A}\mathbf{x} + \mathbf{a}\| > \sqrt{\alpha} (|\mathcal{D}| - 1)$ , which is depicted in Figure 5. This inequality forms a paraboloid with minimum point in  $x = -A^{-1}\mathbf{a}$  and curvature specified by its Hessian  $(A^T A)^{1/2}$ . This Hessian matrix has maximum eigenvalue  $\lambda_{\max} = (|\mathcal{D}| - 1)$  with associated eigenvector  $v_{\max} = \frac{\mathbf{1}}{\|\mathbf{1}\|}$ , and the rest of eigenvalues are equal with value  $\lambda_{\min} = 1$ . The level curve corresponding to the value of  $\sqrt{\alpha} (|\mathcal{D}| - 1)$  is therefore an ellipsoid that grows faster in the direction of  $v_{\max}$ . Then, one sufficient condition to satisfy that the mapping  $T$  is always a contraction, is to construct the region  $X$  outside of the ellipsoid. We define  $X$  as the upper halfspace limited by the hyperplane with normal vector  $v_{\max}$  and most distant point  $x = A^{-1}\mathbf{a} + \sqrt{\alpha} \frac{\mathbf{1}}{\|\mathbf{1}\|}$ . Such region is defined in equation (18) and, since  $X$  is convex, the euclidean projection of  $T$  is well defined. This proves *b)*.

The convergence property of the iterative fixed-point equation directly follows from the Banach fixed point theorem [21, Th.11.1.6.], which states that the mapping admits a unique fixed point  $\mathbf{x}^* = T(\mathbf{x}^*)$  in  $\mathbf{x} \in X$ , and that such point can be found as the limit point of the sequence  $\mathbf{x}^{k+1} = T(\mathbf{x}^k)$ , for any starting point  $\mathbf{x}^0 \in X$ . This proves *c)*.

Finally, to prove *d)*, it is sufficient to show that the following problem **P** is feasible:

$$\begin{aligned} \text{find } & \mathbf{x} \\ \text{P: s.t. } & \mathbf{x} = \sqrt{\alpha} \frac{\mathbf{a} + \mathbf{A}\mathbf{x}}{\|\mathbf{a} + \mathbf{A}\mathbf{x}\|} \\ & \mathbf{1}^T \mathbf{x} + \frac{1}{|\mathcal{D}| - 1} \mathbf{a}^T \mathbf{1} - \sqrt{\alpha |\mathcal{D}|} \geq 0, \end{aligned}$$

which would prove the existence of a point that solves (15) and is inside region  $X$ . An equivalent formulation of **P** is

$$\begin{aligned} \text{find } & \mathbf{x}, z \\ \text{s.t. } & \sqrt{\alpha} \mathbf{1}^T (\mathbf{a} + \mathbf{A}\mathbf{x}) + \left( \frac{1}{|\mathcal{D}| - 1} \mathbf{a}^T \mathbf{1} - \sqrt{\alpha |\mathcal{D}|} \right) z \geq 0 \\ & \|\mathbf{A}\mathbf{x} + \mathbf{a}\| = z, \quad z > 0. \end{aligned}$$

If  $\frac{1}{|\mathcal{D}| - 1} \mathbf{a}^T \mathbf{1} - \sqrt{\alpha |\mathcal{D}|} = 0$  the previous problem is unconditionally feasible. Otherwise, it can be transformed as follows

$$\begin{aligned} \text{find } & \mathbf{y}, z \\ \text{s.t. } & z \geq \frac{-\sqrt{\alpha}}{\frac{1}{|\mathcal{D}| - 1} \mathbf{a}^T \mathbf{1} - \sqrt{\alpha |\mathcal{D}|}} \mathbf{1}^T \mathbf{y} \end{aligned} \quad (26)$$

$$\|\mathbf{y}\| = z, \quad z > 0. \quad (27)$$

Equation (26) represents a halfspace defined by a non vertical and non horizontal hyperplane that passes through the origin. Equations in (27) represent a positive cone. Since the intersection of these spaces is necessarily nonempty the problem is feasible, and so is **P**. This proves *d)* and concludes the proof of the theorem. ■

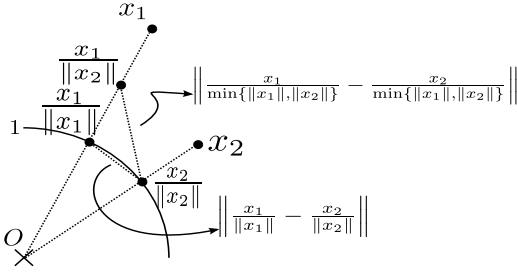


Fig. 4: Graphical interpretation of Lemma 3

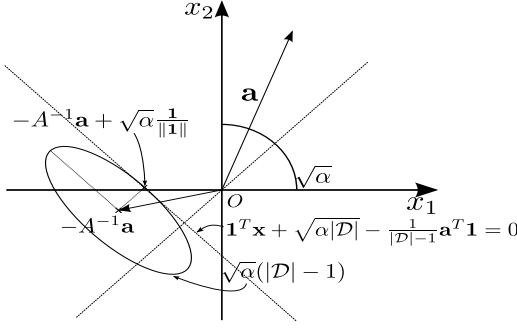


Fig. 5: Graphical interpretation of Theorem 3

## ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their comments, which helped to improve the manuscript.

## REFERENCES

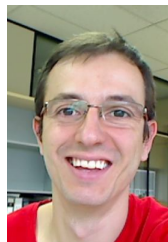
- [1] D.P. Bertsekas and J.N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Athena scientific optimization and computation series. Athena Scientific, 1997.
- [2] I. Atzeni, L.G. Ordonez, G. Scutari, D.P. Palomar, and J.R. Fonollosa, "Demand-side management via distributed energy generation and storage optimization," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 866–876, June 2013.
- [3] K. Cumanan, R. Krishna, V. Sharma, and S. Lambotharan, "Robust interference control techniques for multiuser cognitive radios using worst-case performance optimization," in *Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, USA, Oct. 2008.
- [4] I. Atzeni, L.G. Ordonez, G. Scutari, D.P. Palomar, and J.R. Fonollosa, "Day-ahead bidding strategies for demand-side expected cost minimization," in *IEEE Third International Conference on Smart Grid Communications (SmartGridComm)*, Tainan, Nov 2012, pp. 91–96.
- [5] P. Samadi, A.-H. Mohsenian-Rad, R. Schober, V.W.S. Wong, and J. Jatskevich, "Optimal real-time pricing algorithm based on utility maximization for smart grid," in *International Conference on Smart Grid Communications (SmartGridComm)*, Gaithersburg, MD, Oct 2010, pp. 415–420.
- [6] A.J. Conejo, J.M. Morales, and L. Baringo, "Real-time demand response model," *IEEE Transactions on Smart Grid*, vol. 1, no. 3, pp. 236–242, Dec 2010.
- [7] W. Saad, Z. Han, H.V. Poor, and T. Basar, "Game-theoretic methods for the smart grid: An overview of microgrid systems, demand-side management, and smart grid communications," *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 86–105, Sept. 2012.
- [8] W. Tushar, B. Chai, C. Yuen, D. B. Smith, and H. V. Poor, "Energy management for a user interactive smart community: A Stackelberg game approach," in *2014 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA)*, May 2014, pp. 709–714.
- [9] Y. Wang, W. Saad, N. B. Mandayam, and H. V. Poor, "Integrating energy storage into the smart grid: A prospect theoretic approach," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 7779–7783.
- [10] A.-H. Mohsenian-Rad, V.W.S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, "Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid," *IEEE Transactions on Smart Grid*, vol. 1, no. 3, pp. 320–331, Dec. 2010.
- [11] G. Scutari, D.P. Palomar, F. Facchinei, and J.-S. Pang, "Monotone games for cognitive radio systems," in *Distributed Decision Making and Control*, vol. 417 of *Lecture Notes in Control and Information Sciences*, pp. 83–112. Springer, Jan. 2012.
- [12] J. Wang, G. Scutari, and D.P. Palomar, "Robust MIMO cognitive radio via game theory," *IEEE Transactions on Signal Processing*, vol. 59, no. 3, pp. 1183–1201, 2011.
- [13] J.B. Hiriart-Urruty, "Global optimality conditions in maximizing a convex quadratic function under convex quadratic constraints," *Journal of Global Optimization*, vol. 21, no. 4, pp. 443–453, Dec. 2001.
- [14] D.P. Bertsekas, *Nonlinear programming*, Athena Scientific, Belmont, Mass., 2nd edition, Sept. 1999.
- [15] F. Facchinei and J.-S. Pang, "Nash equilibria: the variational approach," in *Convex optimization in signal processing and communications*. Cambridge University Press, Dec. 2009.
- [16] F. Facchinei, J.-S. Pang, and G. Scutari, "Non-cooperative games with minmax objectives," *Computational Optimization and Applications*, vol. 59, no. 1-2, pp. 85–112, Mar. 2014.
- [17] F. Facchinei and C. Kanzow, "Generalized Nash equilibrium problems," *Annals of Operations Research*, vol. 175, no. 1, pp. 177–211, Mar. 2010.
- [18] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge university press, Mar. 2004.
- [19] K. Nishimura and J. Friedman, "Existence of Nash equilibrium in n person games without quasi-concavity," *International Economic Review*, vol. 22, no. 3, pp. 637–648, Oct. 1981.
- [20] R.A. Horn and C.R. Johnson, *Matrix analysis*, Cambridge University Press, Dec. 2012.
- [21] K.R. Davidson and A.P. Donsig, *Real Analysis and Applications: Theory in Practice*, Springer, Oct. 2009.



wireless networks and smart grid.



to distributed algorithms, optimization and game theory. Since 1990 he has (co)authored more than 30 journal papers and about 150 conference papers. He has also led many private and national Spanish projects and has participated in five European projects, leading three of them.



**Javier Zazo** (S'12) received his Telecommunications Engineer degree from Universidad Politécnica de Madrid (UPM) and Technische Universität Darmstadt (TUD) in 2010, and M.Sc. by Research from the National University of Ireland, Maynooth (NUIM) in 2012. He is currently pursuing a Ph.D. degree in Communications Systems at UPM, funded by an FPU doctoral grant from the Spanish Ministry of Science and Innovation. His current research interests include distributed optimization, game theory and variational inequalities, applied in areas such as

**Santiago Zazo** (M'12) is Telecom Engineer and Dr. Engineer by Universidad Politécnica de Madrid (UPM), since 1990 and 1995 respectively. From 1991 to 1994 he was with the University of Valladolid, and with the University Alfonso X El Sabio, from 1995 to 1997. In 1998, he joined UPM as Associate Professor in Signal Theory and Communications. His main research activities are in the field of Signal Processing, with applications to audio, radar, communications, MIMO and Wireless Sensor Networks. His main activity at this moment is related

**Sergio Valcarcel Macua** (S'10) received his B.S. in Telecommunications Engineering and M.S. in Electronic Systems from Universidad Politécnica de Madrid (UPM), where he is currently a Ph.D. candidate. Before starting his Ph.D., he has been Research Associate at Robotics Institute, Carnegie Mellon University, and Research Visitor at University of California, Los Angeles, and at INRIA Nord, Lille. His research interests include reinforcement learning, optimization, machine learning and statistical signal processing in distributed multiagent systems.